Gilbert DALMASSO - Gilles ANDRIVET

DEUX PROGRAMMES D'EDITION DE MUSIQUE

A quoi serviraient des moyens de calcul très rapides si la transcription des résultats d'un programme devait se faire à la main, avec les risques d'erreurs et la perte de temps conséquents ?

Un problème classique qui se pose aux musiciens-informaticiens débutants est de préférer aux longs et fastidieux transcodages (passage de colonnes de chiffres imprimés à une notation sur portée) l'impression d'une partition sur portée directement interprétable. Le résultat, parfois un peu incomplet étant donné les limites de la machine à écrire, peut alors être complété à peu de frais (lignes des portées, crescendos etc.).

Nous proposons ici deux programmes destinés à faciliter l'abord de ces problèmes : le premier (SORTIE 1 SUR PORTEE, G.DALMASSO), facilement abordable, montre le mécanisme simple utilisable pour imprimer une portée à partir d'un vecteur contenant, sous forme codée, une mélodie écrite en valeurs égales, en gérant soi-même les buffers de sortie ; le second (PROGRAMME D'ECRITURE MUSICALE, G.ANDRIVET), un peu plus compliqué, résout de surcroît les problèmes de durées et de superposition des voix.

-::-::-::-::-

- SORTIE 1 SUR PORTEE -

On lit la marge (MARGE) à partir de laquelle la portée doit être imprimée. On lit le vecteur contenant la mélodie codée (MEL).

On range dans un tableau de travail (TRAV) l'équivalent du contenu de MEL sous forme de notes identiques rangées selon leurs hauteurs (une note par ligne du tableau TRAV, la colonne remplie correspondant à la hauteur de la note).

On écrit les repères de portées sous forme de lettres 'T' superposées.

On passe de TRAV au buffer de sortie BUF selon la marge choisie (à chaque fois une ligne de TRAV est transférée dans BUF et imprimée).

L'extension à plusieurs mélodies se fait sans difficultés (voir le programme SORTIE 2 SUR PORTEES). De la même façon, on pourra faire un programme d'édition pour un nombre variable de voix (dans les limites de la taille du papier), le nombre de voix pouvant être mis en données.

-::-::-::-::-

- PROGRAMME D'ECRITURE MUSICALE -

Aux Musiciens.

Ce programme prétend aligner sur une portée traditionnelle un certain nombre de notes. Il tient à conserver la représentation classique et scolaire d'une succession de sons... Aussi mauvaise soit-elle !

Aux Informaticiens.

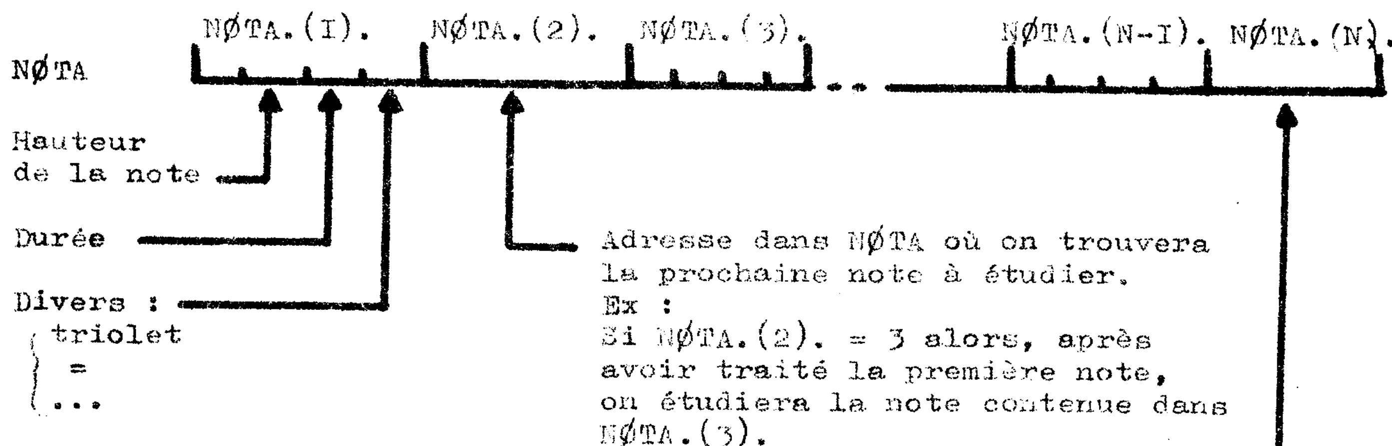
Le but n'est pas de pondre un programme performant. La place mémoire ne manque pas, elle est largement utilisée pour faciliter la compréhension du listing.

-::-::-::-::-

Les Entrées.

Il n'est pas question de faire entrer des alpha du type " DO "," NOIRE ", ou " TRIOLET " dans la machine. Surtout en Algol 60. Les chiffres seront donc utilisés en données. Comment les organiser en mémoire ?

Deux tableaux NØTA et NØTB(si on veut sortir deux portées), qui auront une structure de ce modèle :





La seconde noire de la deuxième portée devrait coïncider avec la première croche du triolet.

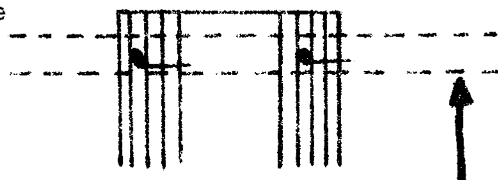
Même représentation pour NØTB.
La note s'écriera donc de la façon suivante :
Ex : 042I_3
Soit
04 ....c'est un FA
2 ....c'est une noire
I ....c'est un dièse
3 ....la note suivante est en NØTA.(3).

Les Sorties.

La machine à écrire n'étant pas adaptée à l'écriture musicale, il a fallu user de certains stratagèmes.

- Ecriture verticale des portées.

Ce qui implique, sur une même ligne, l'écriture d'un fragment de chaque portées



Sens de déroulement du papier.

- Ecriture en rouge des dieses.
- Utilisation d'une mémoire tampon, qui chargera une ligne complète et la videra sur le papier.

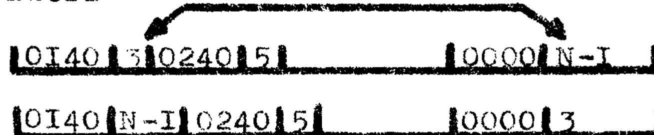
Les Déclarations.

On ne lésine pas sur les paramètres.

-A;B;C;D-
La note est donc un nombre dont seuls les composants nous intéressent. Il faut les décoder.
En reprenant l'exemple du 042I_3 , les valeurs respectives de hauteur, durée, divers, adresse de la note suivante, seront retrouvées par la procédure ABC.

A = Hauteur
B = Durée
C = Divers
D = Adresse de la note suivante

Une permutation entre les 'D' de la note en lecture et de la dernière note (0000) du tableau, va être utilisée.
Ce qui donnera
Ex :



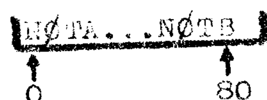
Ayant lu et écrit la note 0I40, on étudiera la note d'adresse N-I soit 0000, donc aucune note écrite.
Par ce jeu de permutation d'adresse, il sera possible de rendre lisible et correct les écritures musicales.

Le paramètre K indiquera s'il y a eu déjà permutation :
Si K = 0 Aucune permutation
Si K = I Permutation sur NØTA
Si K = 2 Permutation sur NØTB

K sera testé avant toute attaque sur une note suivante.

-M,N-
Compteurs, respectivement de NØTA et NØTB, qui comptabilisent le nombre de temps dans chaque portée.
Si M est supérieur à N, il faudra faire une permutation du type étudié ci-dessus, sur NØTA, de façon à rechercher l'équilibre M égal N.
Idem si N supérieur à M.

-Y-
Paramètre utilisé dans le remplissage du buffer.
Ce buffer est rempli par

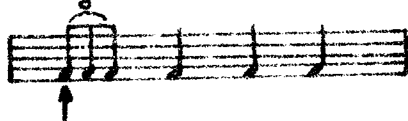


Y varie de O à la moitié du buffer pour NØTA, l'autre moitié pour NØTB.

-X-
Compteur permettant de savoir si on travaille sur NØTA ou sur NØTB.

-Z-
Paramètre d'incrémentation dans un certain nombre de boucles 'FOU'. Utilisé aussi pour indicer une impression en rouge.

-E,F-
Pour définir ces paramètres, il faut préciser l'étude des triolets.
Mathématiquement 3=2 est surprenant. Il suffit d'affecter à la première note d'un triolet un coéficient C=2, qui impliquera un saut dans l'addition des temps dans la mesure.
Ex :



Dans la mesure, la valeur en temps de cette note sera considérée comme nulle : on retrouvera alors 2=2.
Ici intervient E (pour NØTA); un triolet sans le petit '3', n'est pas un triolet; test sur E; c'est à dire sur le C de la note précédente.
En quelque sorte E:=C si C=2
à la note suivante test sur E;
Idem pour F.

-I,J-
L'incrémentation, ou l'avance sur NØTA étant différente de celle de NØTB, I et J seront respectivement les indices de ces tableaux.

Les Procédures.

ZOZO et PRINT.
Ouverture et fermeture d'impression en rouge. Suivant la valeur de C, on appelle ZOZO avec Z:=I.
Suivant la valeur de Z, on appelle PRINT, avec Z:=O.

PLUS et TEST
Procédures d'incrémentation de M et N , d'après la valeur de C .
Si C=2 première note d'un triolet : aucune incrémentation.

ABC
Procédure qui décode les significatifs de chaque note.

ECR
En appelant ECR, on vide le buffer suivant son contenu numérique, en transformant ces chiffres en caractères graphiques.

PORT et TUTU.
Impression du cadrage de la portée sur le papier.

Les Etiquettes.

EI:
Départ de l'étude et écriture d'une ligne complète.

E2:
Renvoi pour le décodage, sur une même ligne, de la deuxième portée.

E6: EIO: et EII:
Etiquettes facilitant une série de test sur X et C.

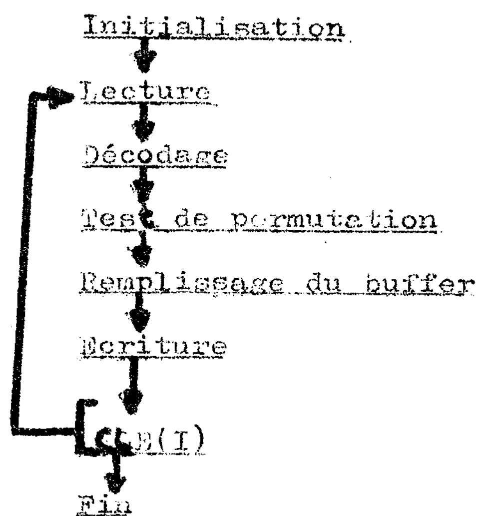
E7:
Utilisé pour sauter la permutation dans NØTA ou NØTB si K:=0.

E8:
Evite la permutation dans NØTA ou NØTB si N=M.

E9:
Etiquette de fin de programme si CLE(I).

-::-::-::-::-::-

ORGANIGRAMME DES FONCTIONS



```
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
x                                x
x       SORTIE 1 SUR PORTEE      x
x                                x
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

'DEB'
'COM'SORTIE1;

'ENT'I,J,K,MARGE;
'ENT''TAB'MEL.(1:30).,BUF.(1:70).,TRAV.(1:30,1:8).;

'COM'LECTUREDESDONNEES;
LIRC(MARGE);LIRTC(MEL);

'COM'INITIALISATIONDESTABLEAUX;
'POU'I:=1'PAS'1'JUS'30'FAI'
'POU'J:=1'PAS'1'JUS'8'FAI'
TRAV.(I,J).:=0;
'POU'J:=1'PAS'1'JUS'70'FAI'
BUF.(J).:=0;

'COM'PASSAGEDEMELATRAV;
'POU'I:=1'PAS'1'JUS'30'FAI'
TRAV.(I,MEL.(I).).:=1;

'COM'ECRITUREDESREPERES;
'POU'K:=MARGE+1'PAS'2'JUS'MARGE+9'FAI'
BUF.(K).:=3;
'POU'K:=1'PAS'1'JUS'70'FAI'
'SI'BUF.(K).'EG'3'ALO'EXL(°T@)'SIN'EXL(° @);IMPR;

'COM'PASSAGEDETRAVABUFETIMPRESSION;
'POU'I:=1'PAS'1'JUS'30'FAI''DEB'
'POU'J:=1'PAS'1'JUS'70'FAI'
BUF.(J).:=0;
'POU'J:=1'PAS'1'JUS'8'FAI'
'SI'TRAV.(I,J).'EG'1'ALO'BUF.(MARGE+J).:=1;
'POU'K:=1'PAS'1'JUS'70'FAI'
'SI'BUF.(K).'EG'1'ALO'EXL(°O@)'SIN'EXL(° @);
IMPR;
'FIN';'FIN'#

xxxxxxxxxxxxxxxxxxxx
x     DONNEES      x
xxxxxxxxxxxxxxxxxxxx

20

1 2 5 3 4 6 8 7 5 3 2 5 6 8 7 5 4 2 3 5
6 8 5 4 2 1 3 5 7 8
```

```
***********************************************
*                                             *
*      PROGRAMME D ECRITURE MUSICALE          *
*                                             *
***********************************************
'DEB'
'ENT'A,B,C,D,E,F,I,J,K,M,N,X,Y,Z;

'ENT''TAB'NOTA,NOTB.(1:90).,BUF.(0:40).;

'PRO'PRINT(X1);'ENT'X1;'DEB''SI'X1'EG'1'ALO'
EXL(°$@)'FIN';

'ENT''PRO'PLUS(X1,X2,X3,X4);'ENT'X1,X2,X3,X4;'DEB'
'SI'X1'EG'1'ALO'X2:=X2+X4'SIN'
'SI'X1'EG'2'ALO'X3:=X3+X4'FIN';

'PRO'ABC(X1,X2,X3,X4,X5,X6);'ENT'X1,X2,X3,X4,X5,X6;
'DEB'X1:=X4%X5;X2:=(X4-(X1*X5))%X6;
X3:=X4-((X1*X5)+(X2*X6))'FIN';

'PRO'ECR(X1);'ENT'X1;'DEB'
'SI'X1'EG'0'ALO'EXL(° @)'SIN'
'SI'X1'EG'1'ALO'EXL(°*@)'SIN'
'SI'X1'EG'2'ALO'EXL(°*@)'SIN'
'SI'X1'EG'3'ALO'EXL(°-@)'SIN'
'SI'X1'EG'4'ALO'EXL(°O@)'SIN'
'SI'X1'EG'5'ALO'EXL(°,@)'SIN'
'SI'X1'EG'6'ALO'EXL(°)@)'SIN'
'SI'X1'EG'7'ALO'EXL(°W@)'FIN';

'PRO'PORT;'DEB''ENT'X1;
ESPACE(1);'POU'X1:=1'PAS'1'JUS'5'FAI'
EXL(° T@);ESPACE(8)'FIN';

'PRO'ZOZO(X1);'ENT'X1;
'DEB'EXL(°#@);X1:=1'FIN';

'PRO'TEST(X1,X2,X3,X4,X5,X6);'ENT'X1,X2,X3,X4,X5,X6;
'DEB''SI'X1'EG'0'ALO'PLUS(X2,X3,X4,X5)'SIN'
'SI'X1'EG'1'ALO''DEB'PLUS(X2,X3,X4,X5);ZOZO(X6)'FIN'
'FIN';

E:=F:=N:=M:=K:=0;I:=J:=1;
LIRTC(NOTA);LIRTC(NOTB);PORT;PORT;
EXL(°%@);IMPR;
E1:
'POU'Z:=1'PAS'1'JUS'40'FAI'BUF.(Z).:=0;
'SI'N'EG'8'ET'M'EG'8'ALO''DEB'PORT;PORT;
EXL(°%@);IMPR;N:=M:=0'FIN';
ABC(A,B,C,NOTA.(I).,100,10);
X:=1;Y:=0;
E2:
BUF.(A+Y).:=B;
'POU'Z:=Y+1'PAS'1'JUS'Y+20'FAI''DEB'
```

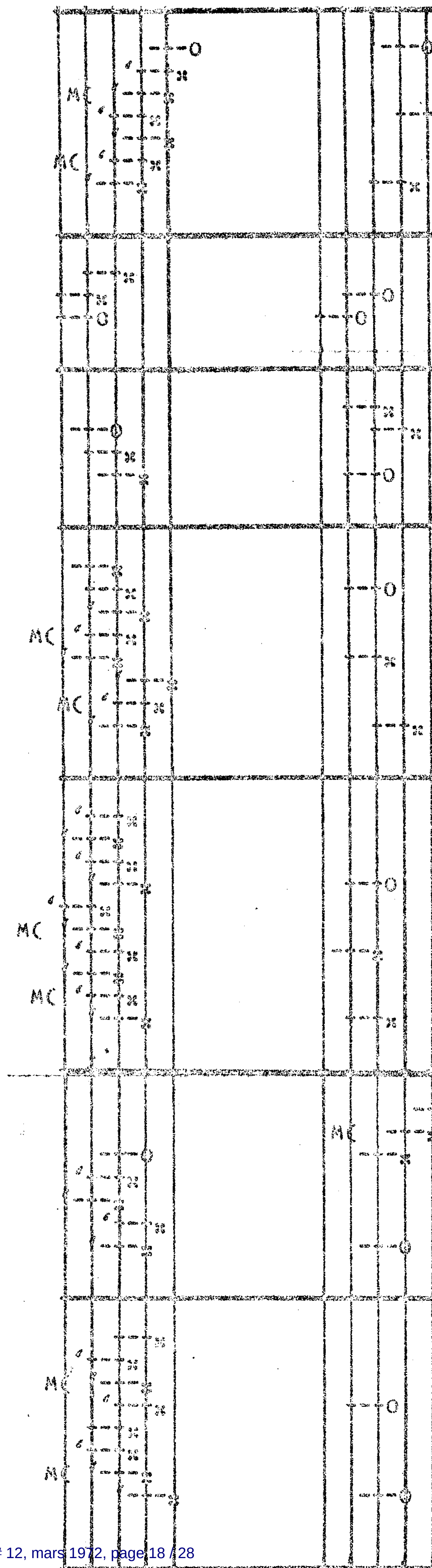
```
'SI'BUF.(Z).'EG'1'OU'BUF.(Z).'EG'2'OU'BUF.(Z).'EG'
4'ALO''DEB'
BUF.(Z+1).:=BUF.(Z+2).:=BUF.(Z+3).:=3'FIN';
'SI'BUF.(Z).'EG'1'ALO'BUF.(Z+4).:=5;
'SI'
(BUF.(Z).'EG'1'OU'BUF.(Z).'EG'2)'ET'Z'INF'33'ET'
((X'EG'1'ET'
(E'EG'2'OU'E'EG'3)))'OU'
(X'EG'2'ET'
(F'EG'2'OU'F'EG'3)))
'ALO''DEB'
BUF.(Z+6).:=6;BUF.(Z+7).:=7;E:=F:=0'FIN''FIN';
'SI'X'EG'2'ALO''ALL'E10;
'SI'C'EG'2'OU'C'EG'3'ALO'
E:=C;'ALL'E11;
E10:
'SI'C'EG'2'OU'C'EG'3'ALO'
F:=C;E11:
TEST(C,X,N,M,B,Z);
'POU'D:=Y+1'PAS'1'JUS'Y+19'FAI'
ECR(BUF.(D).);
X:=X+1;
'SI'X'DIF'2'ALO''ALL'E6;
ABC(A,B,C,NOTB.(J).,100,10);
Y:=20;
PRINT(Z);
'ALL'E2;
E6:
IMPR;
'SI'K'INF'1'ALO''ALL'E7;
'SI'K'EG'1'ALO''DEB'
I:=NOTA.(I+1).-2;NOTA.(I+1).:=I+2;NOTA.(90).:=89'FIN';
'SI'K'EG'2'ALO''DEB'
J:=NOTB.(J+1).-2;NOTB.(J+1).:=J+2;NOTB.(90).:=89'FIN';
K:=0;
E7:
'SI'N'EG'M'ALO''ALL'E8;
'SI'N'SUP'M'ALO''DEB'
D:=NOTA.(I+1).;NOTA.(I+1).:=89;NOTA.(90).:=D;K:=1'FIN';
'SI'M'SUP'N'ALO''DEB'
D:=NOTB.(J+1).;NOTB.(J+1).:=89;NOTB.(90).:=D;K:=2'FIN';
E8:
I:=NOTA.(I+1).;
J:=NOTB.(J+1).;
'SI'CLE(1)'ALO''ALL'E9;
PRINT(Z);'ALL'E1;
E9:
PRINT(Z);
EXL(°  = = = = =            = = = = =@);IMPR
'FIN'#
```

101

```
********************
*      DONNEES     *
********************
```

540 3 641 5 540 7 522 9 320 11 120 13 62
0 15 720 17 640 19 420 21 620 23 640 25
641 27 421 29 620 31 840 33 640 35 420 3
7 220 39 340 41 0 89 0 89 0 89 0 89 0 89
0 89 0 89 0 89 0 89 0 89 0 89 0 89 0 89
0 89 0 89 0 89 0 89 0 89 0 89 0 89 0 89
0 89 0 89 0 89 0 89

312 3 510 5 610 7 621 9 412 11 510 13 61
1 15 420 17 510 19 410 21 710 23 610 25
540 27 512 29 610 31 710 33 612 35 710 3
7 810 39 510 41 610 43 710 45 610 47
512 49 410 51 310 53 712 55 610 57 510 5
9 620 61 720 63 521 65 620 67 740 69
840 71 820 73 620 75 512 77 410 79 310 8
1 412 83 310 85 210 87 140 89 000 89



M
MC
MC
MC
MC
MC
M
M
M